Fig. 1

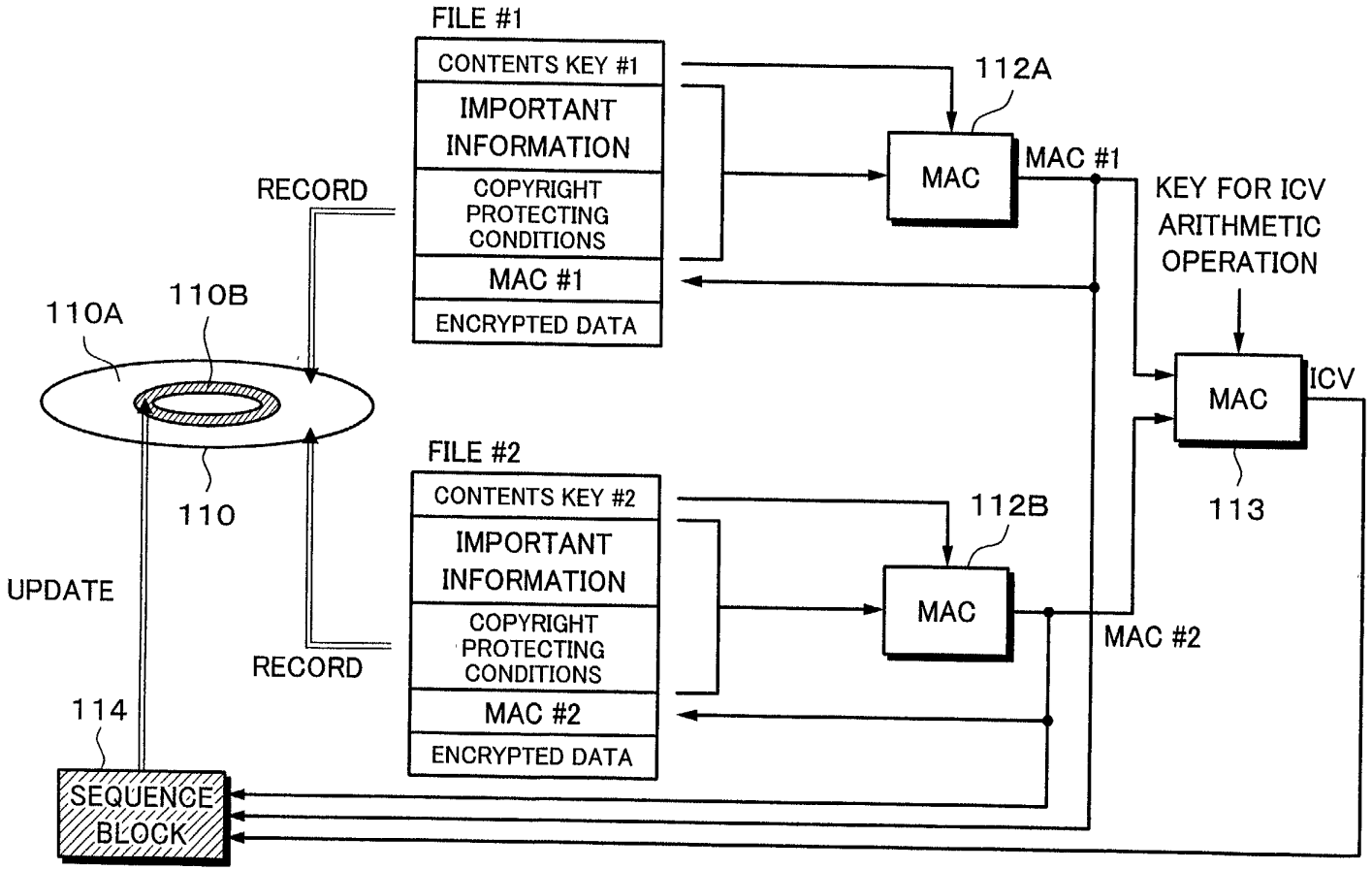


Fig. 2

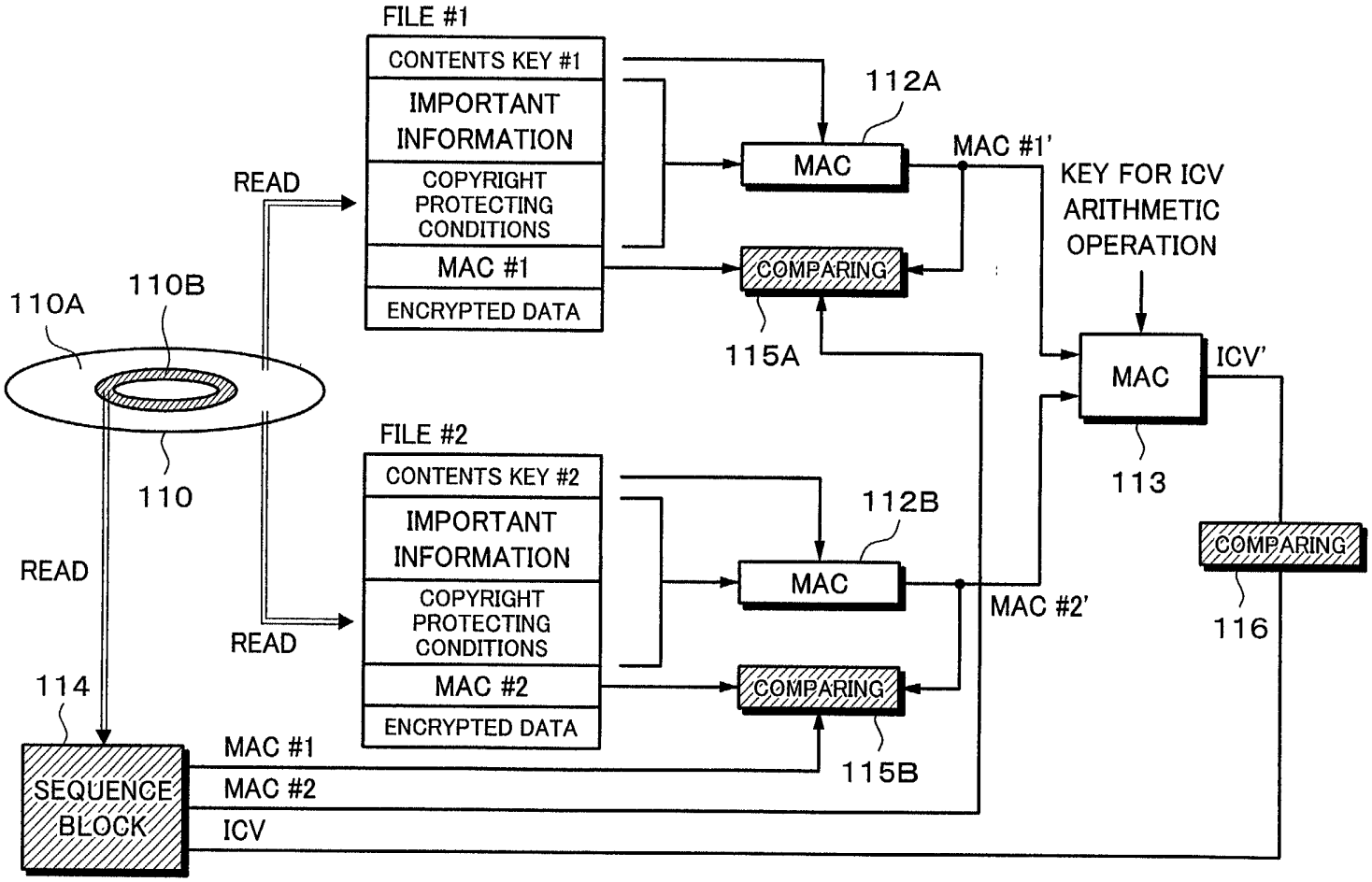


Fig. 3

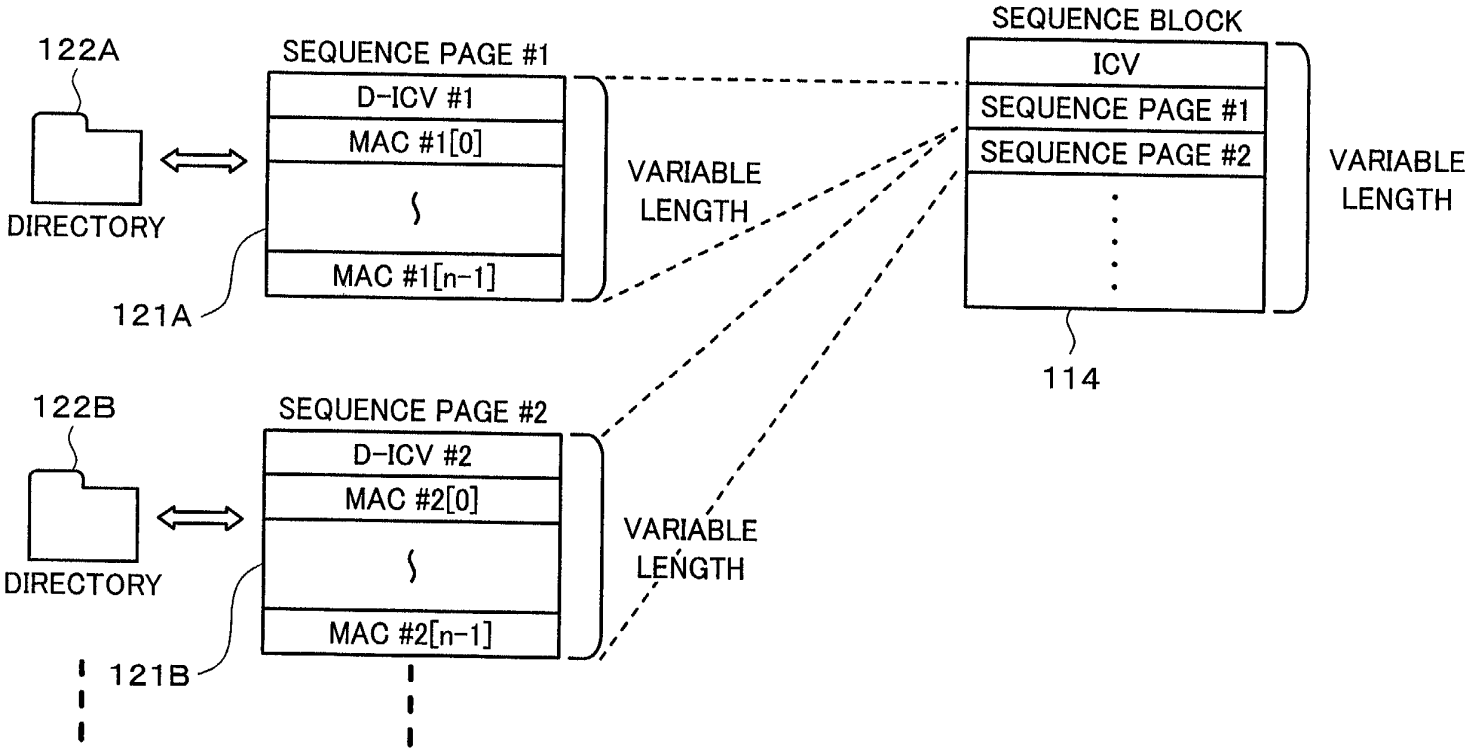


Fig. 4

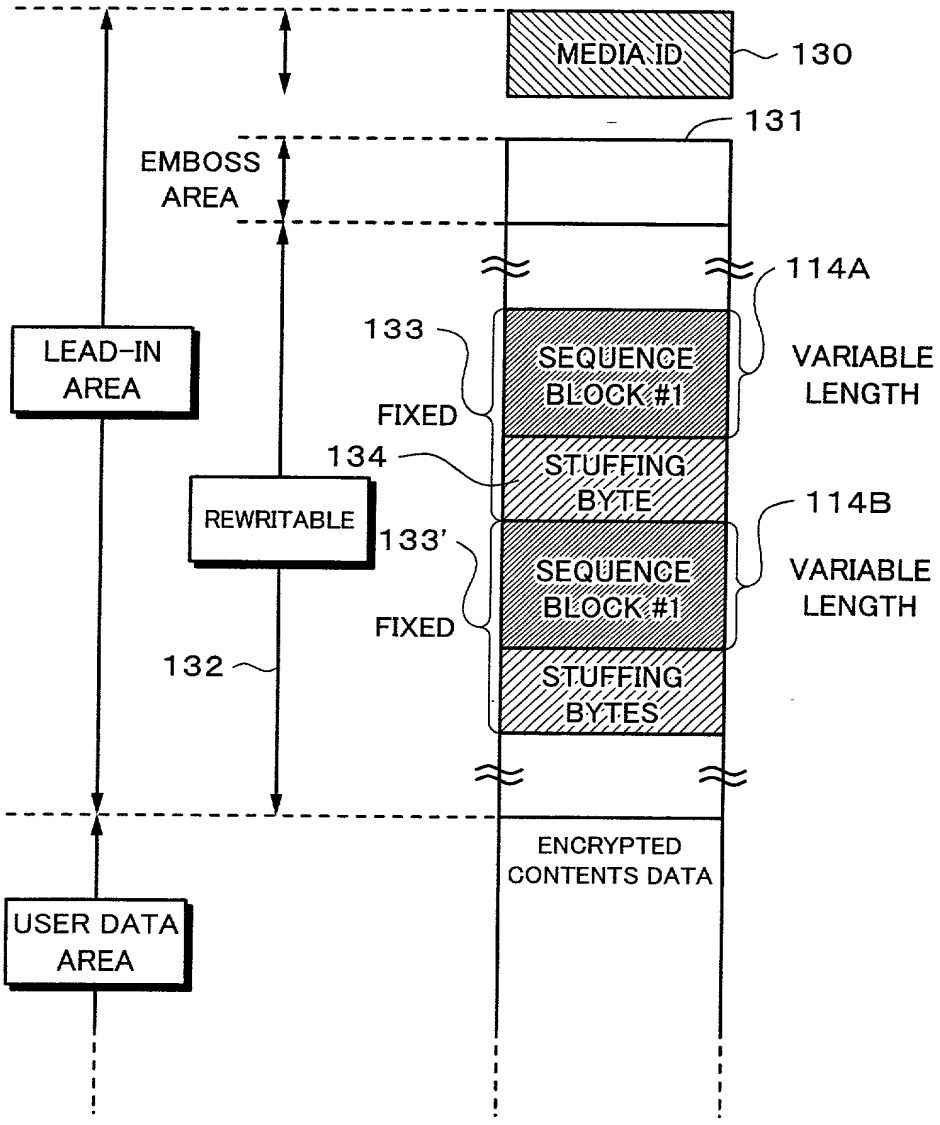
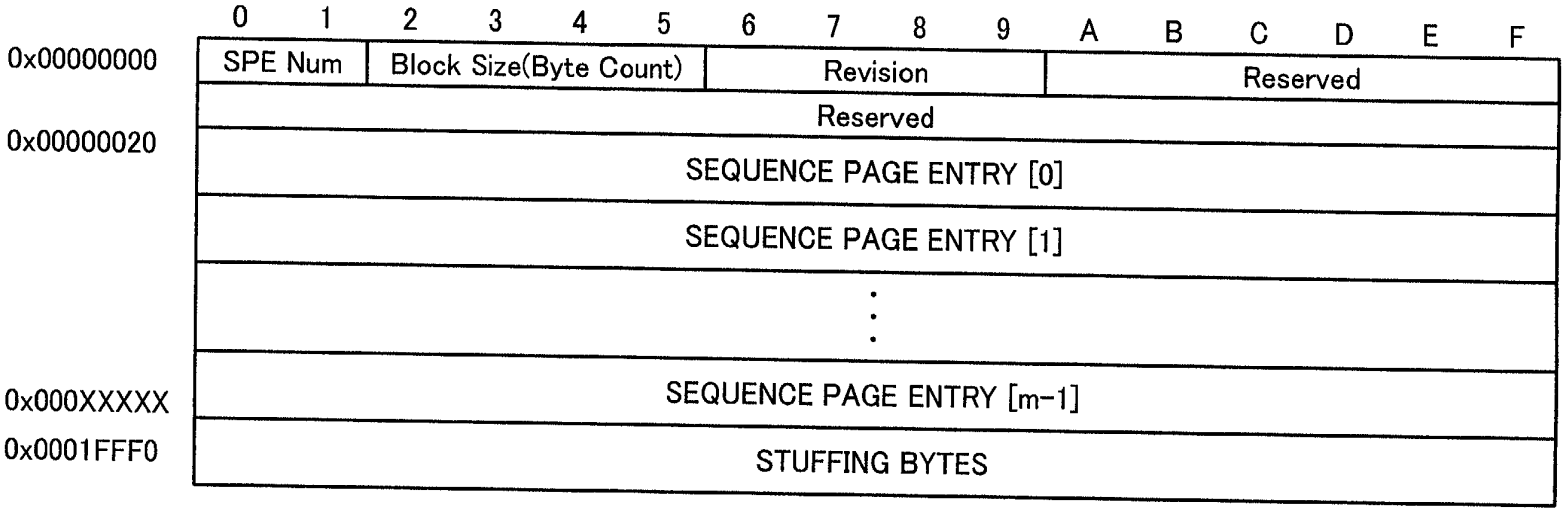

MEDIA ID
130
131
EMBOSS AREA
114A
LEAD-IN AREA
133
SEQUENCE BLOCK #1
VARIABLE LENGTH
FIXED
134
STUFFING BYTE
REWRITABLE
114B
133'
SEQUENCE BLOCK #1
VARIABLE LENGTH
FIXED
132
STUFFING BYTES
ENCRYPTED CONTENTS DATA
USER DATA AREA

Fig. 5

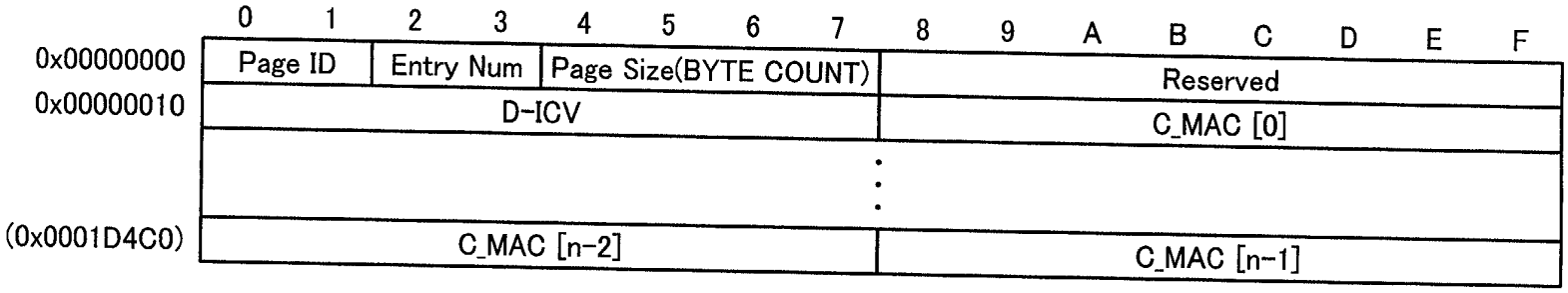


SPE Num : Sequence Page Entry Number
THE TOTAL NUMBER OF ENTRIES OF SEQUENCE PAGE

Block Size : Sequence Block Size
SIZE OF SEQUENCE BLOCK, COUNT THE NUMBER OF BYTES FROM HEAD BYTE TO LAST BYTE OF LAST ENTRY

Revision : Revision Number
THE NUMBER OF TIMES OF REVISION OF SEQUENCE BLOCK, VALID/INVALID STATE
INCREASE BY "1" FROM INITIAL STATE "0"
0xFFFFFFFF = Invalid Number
INDICATES THAT THIS SEQUENCE BLOCK IS INVALID OR IS BEING REVISED

Fig. 6



Page ID : Sequence Page ID
ID FOR ASSOCIATING SEQUENCE PAGE WITH FOLDER

Entry Num : MAC Entry Number
THE TOTAL NUMBER OF ENTRIES

Page Size : Sequence Page Size
SIZE OF SEQUENCE PAGE, COUNT THE NUMBER OF BYTES FROM HEAD BYTE TO LAST BYTE OF LAST ENTRY

C_MAC[n] : Contents MAC Value
MAC VALUE CALCULATES EVERY FILE (CONTENTS)

Fig. 7A Fig. 7B Fig. 7C

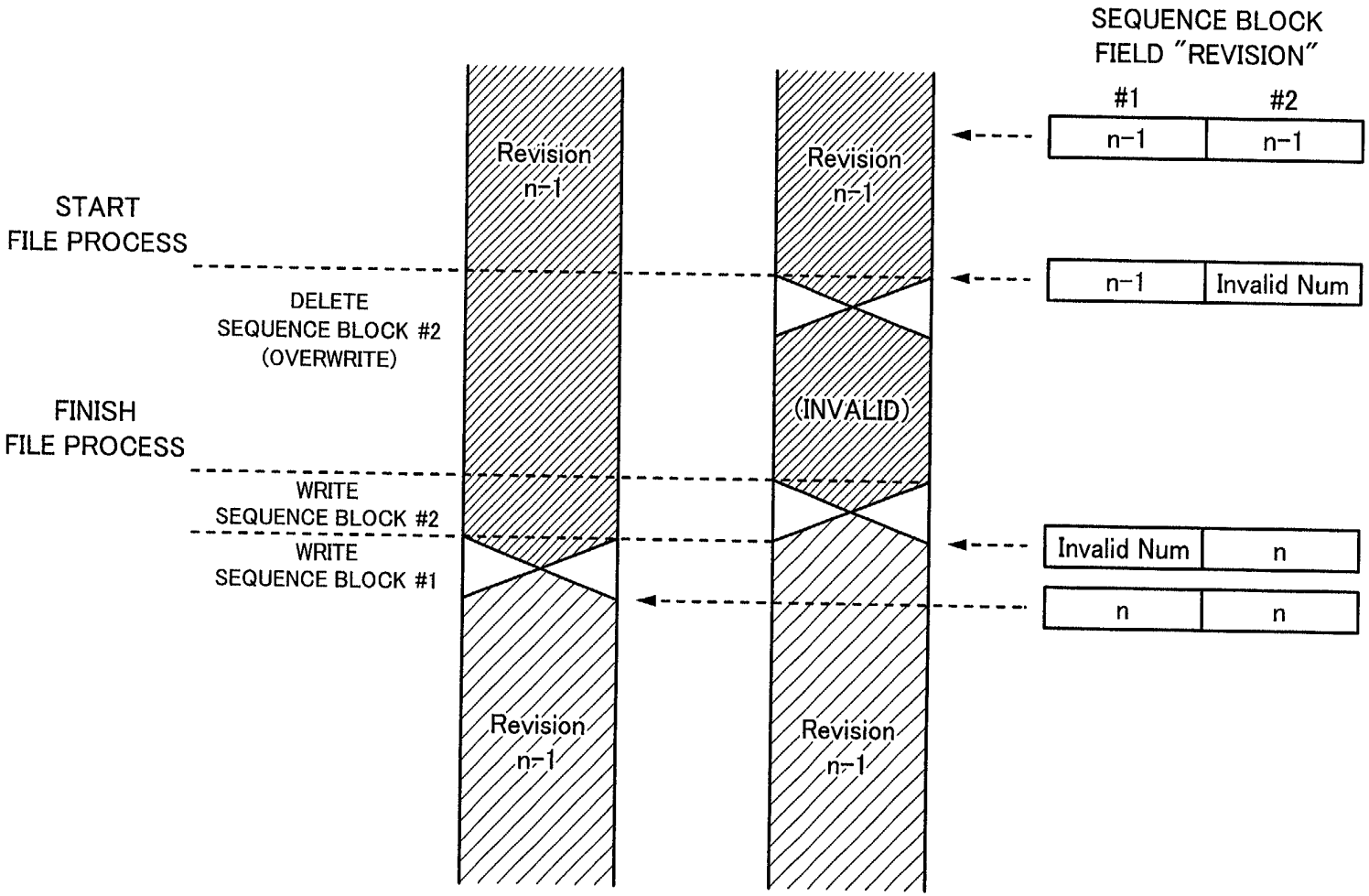


Fig. 8

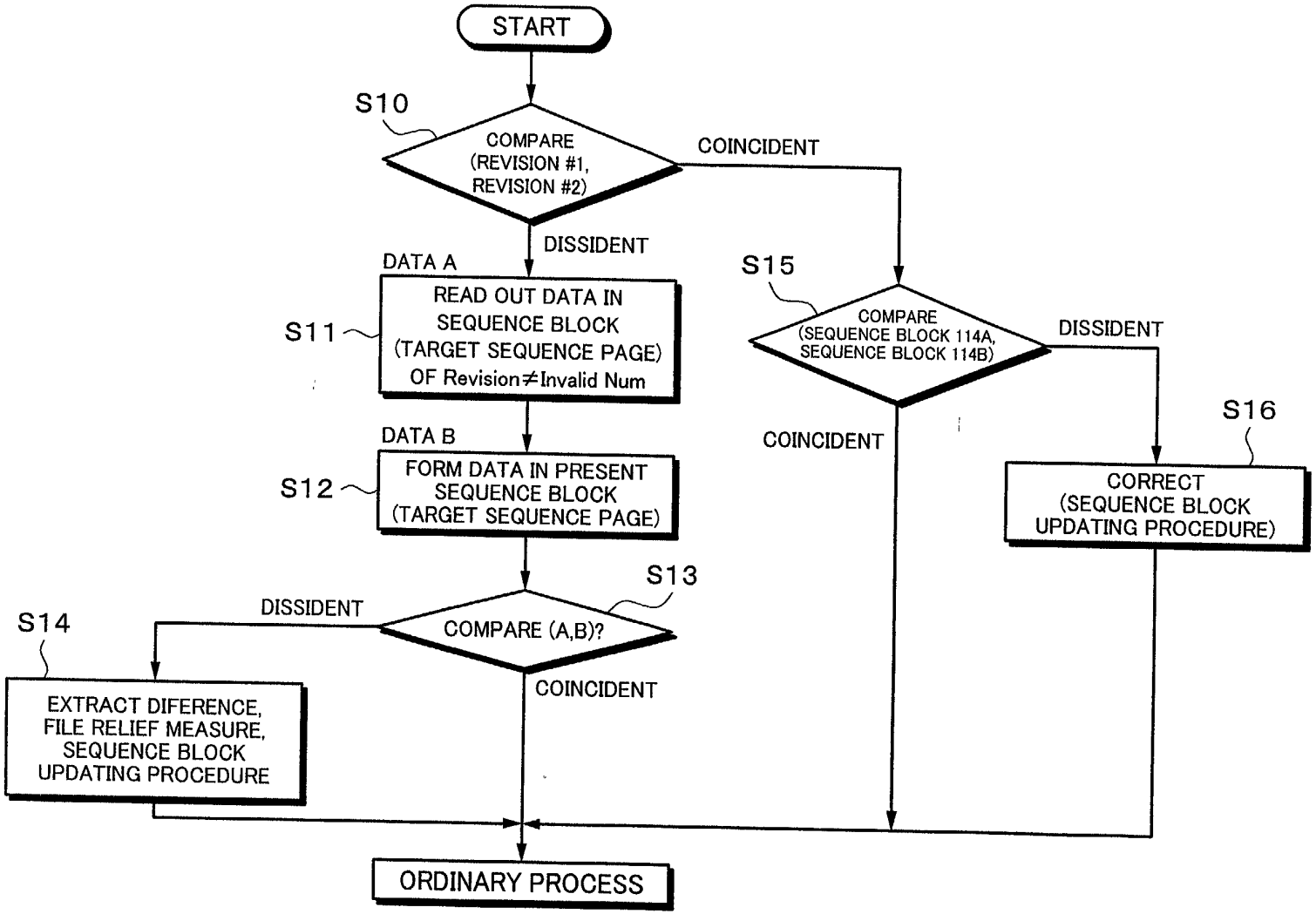

START
S10
COMPARE (REVISION #1, REVISION #2)
COINCIDENT
DISSIDENT
DATA A
S11
READ OUT DATA IN SEQUENCE BLOCK (TARGET SEQUENCE PAGE) OF Revision≠Invalid Num
DATA B
S12
FORM DATA IN PRESENT SEQUENCE BLOCK (TARGET SEQUENCE PAGE)
S13
COMPARE (A,B)?
DISSIDENT
S14
EXTRACT DIFERENCE, FILE RELIEF MEASURE, SEQUENCE BLOCK UPDATING PROCEDURE
COINCIDENT
S15
COMPARE (SEQUENCE BLOCK 114A, SEQUENCE BLOCK 114B)
DISSIDENT
COINCIDENT
S16
CORRECT (SEQUENCE BLOCK UPDATING PROCEDURE)
ORDINARY PROCESS

Fig. 9

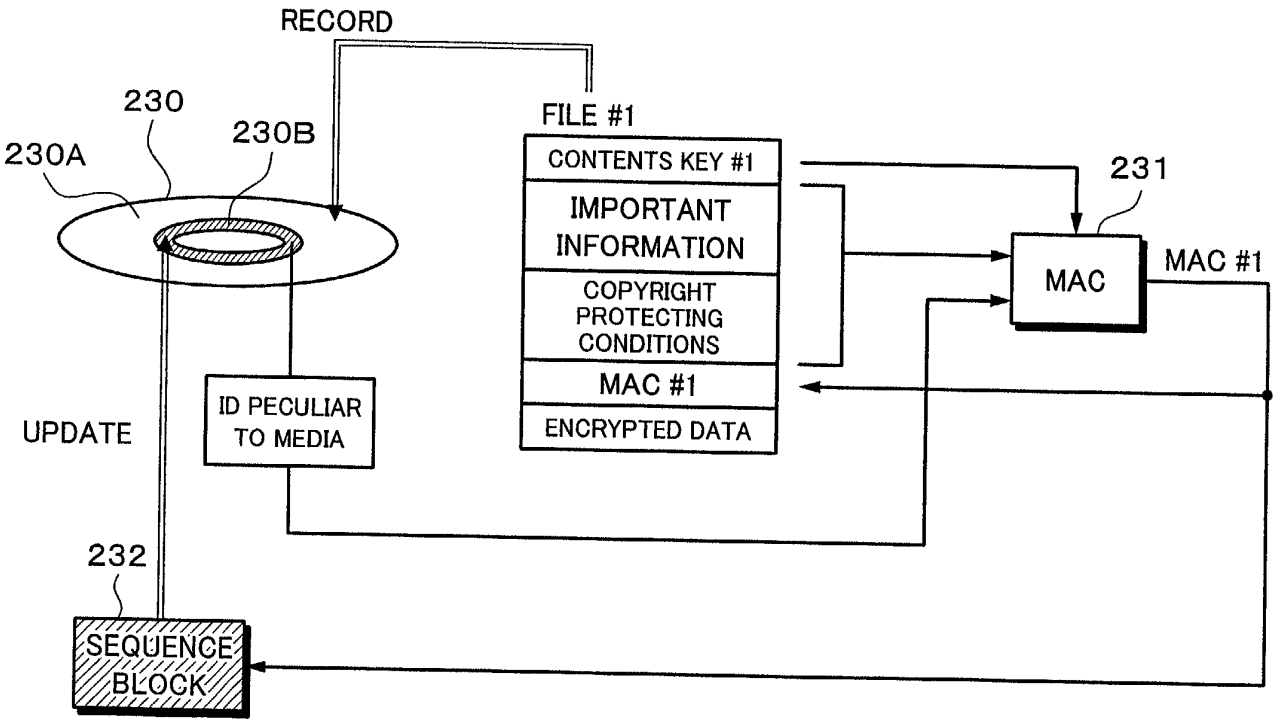


Fig. 10A

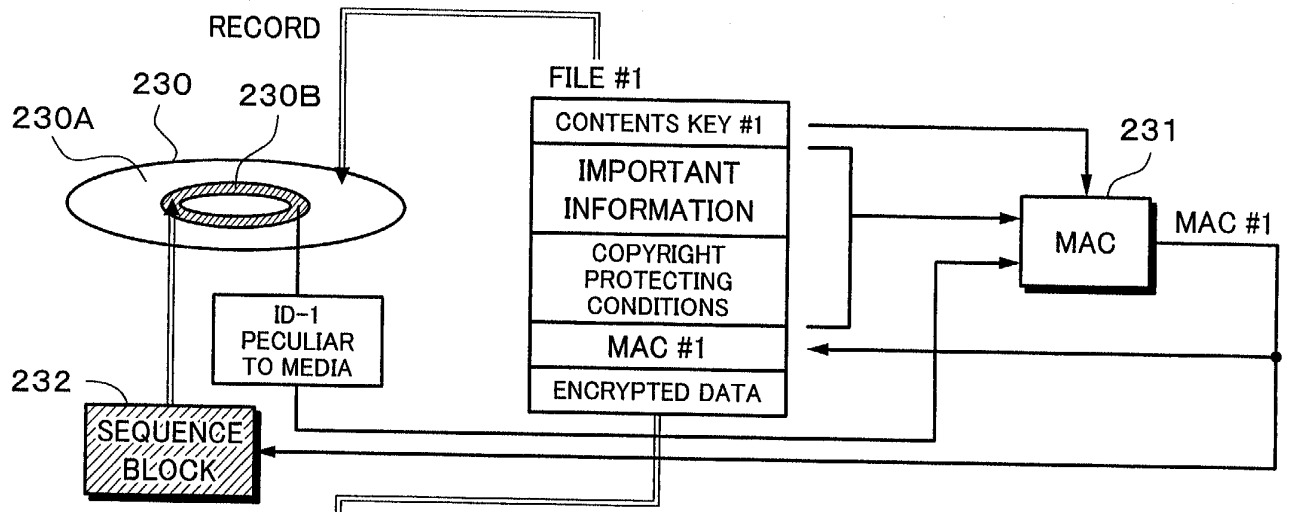


Fig. 10B

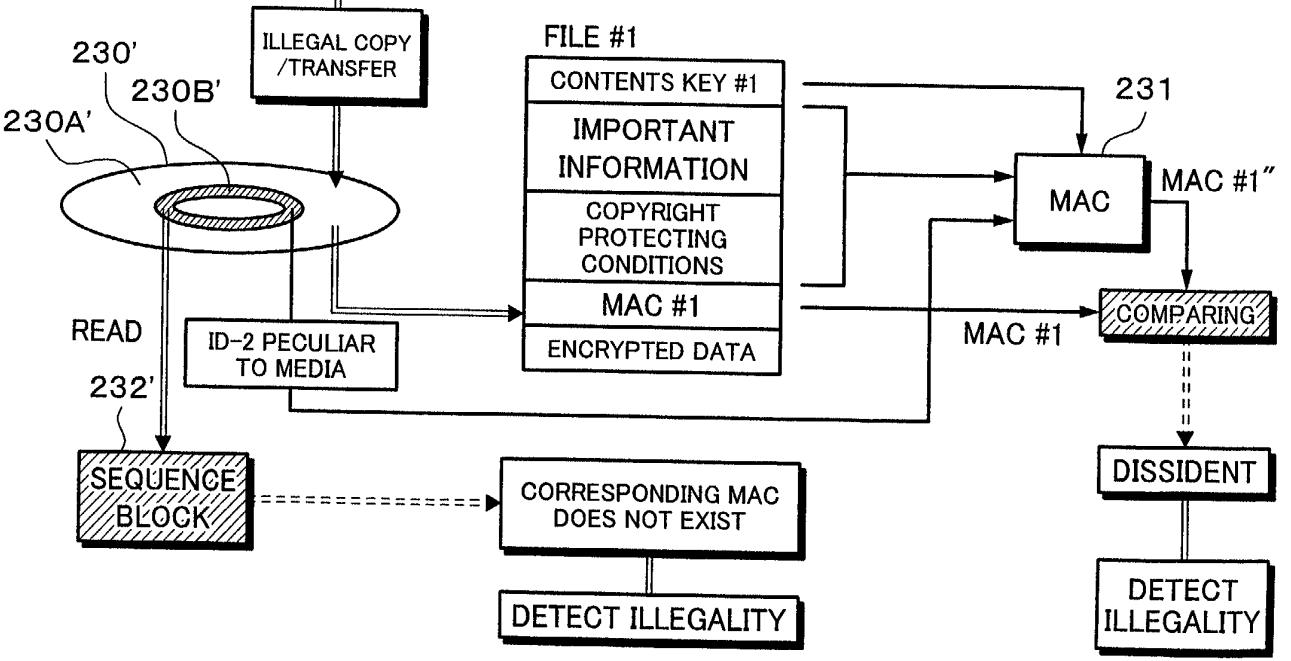


Fig. 11

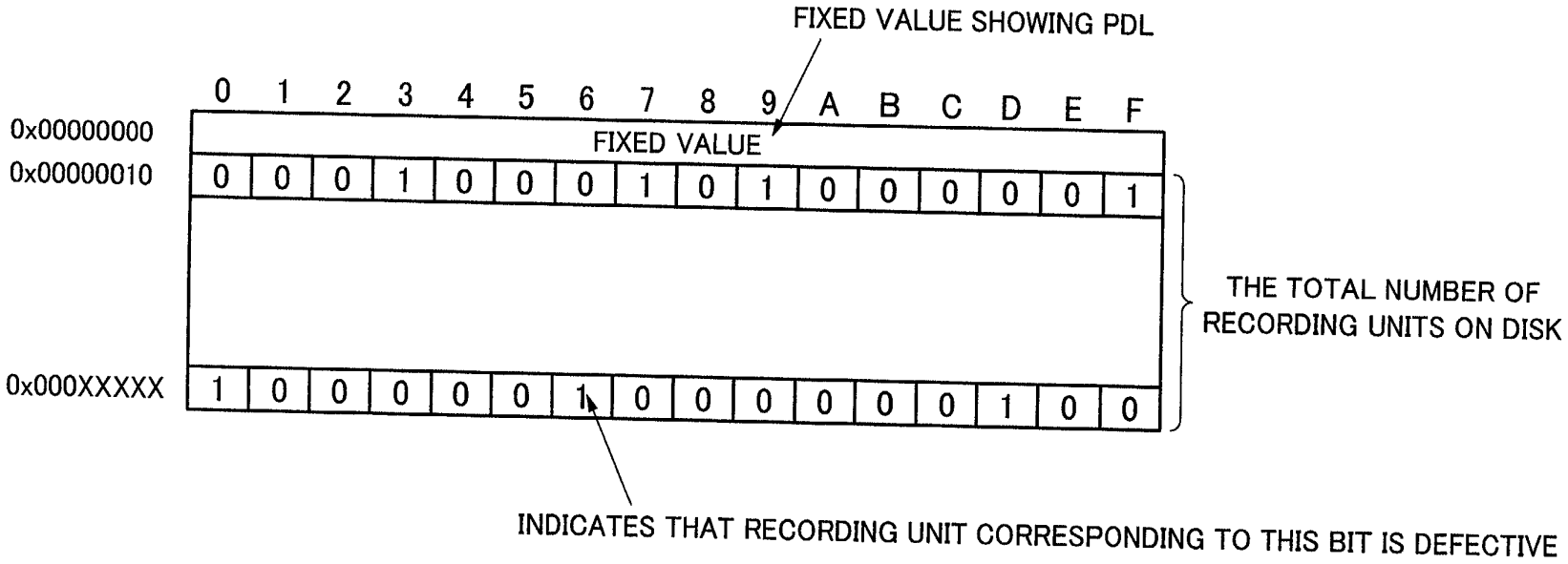


Fig. 12

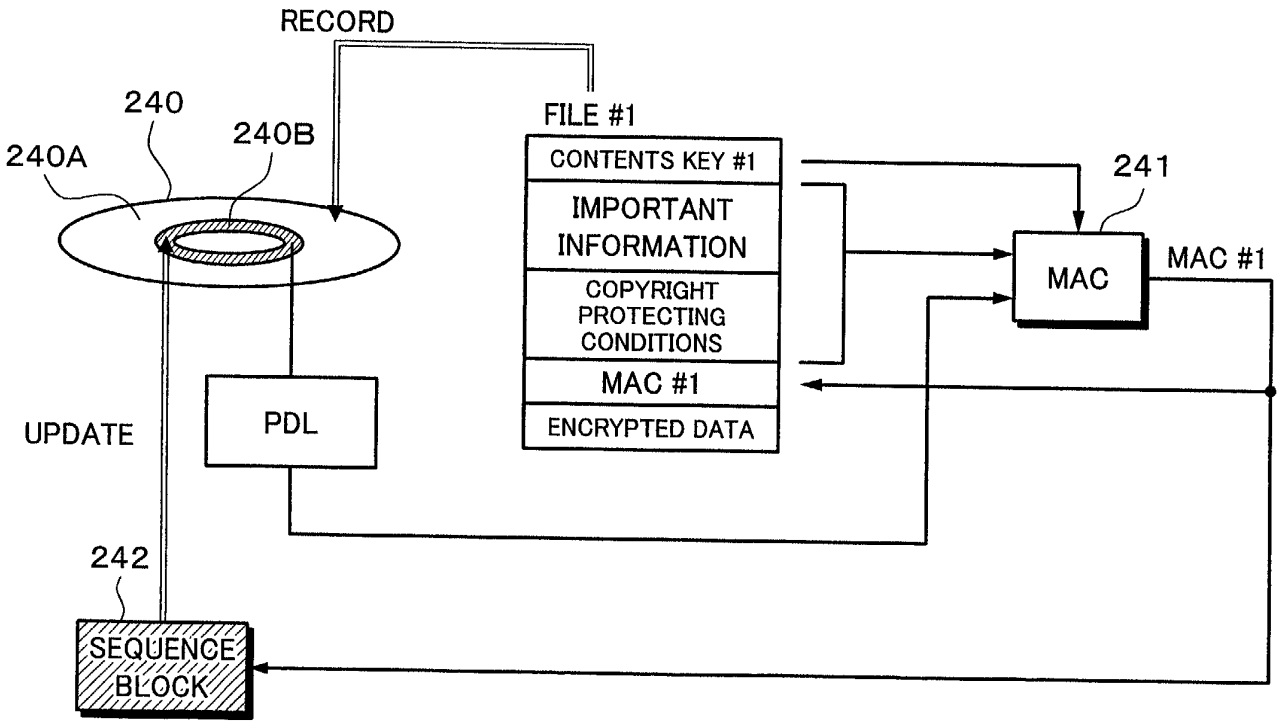


Fig. 13A

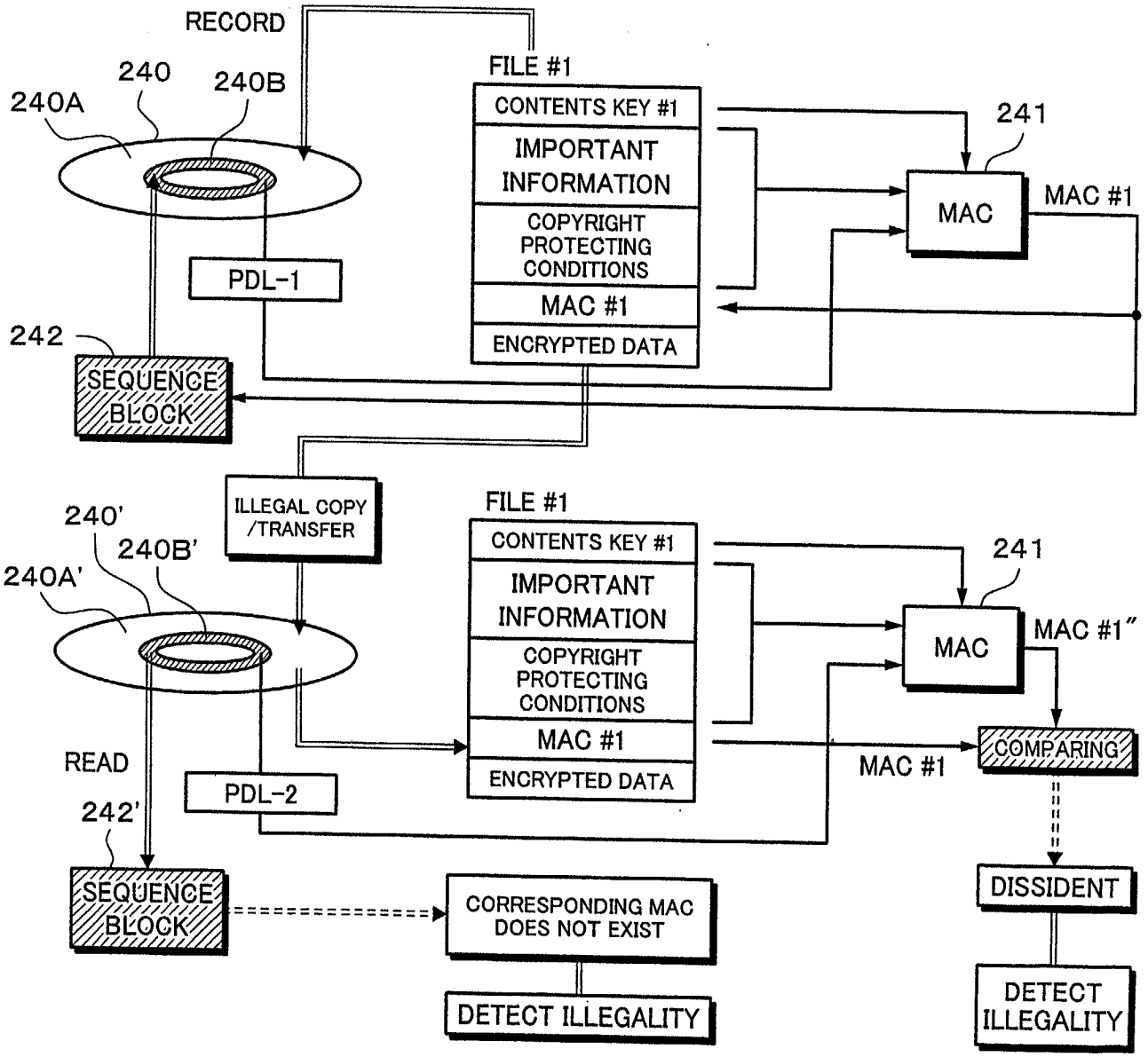


Fig. 13B

Fig. 14

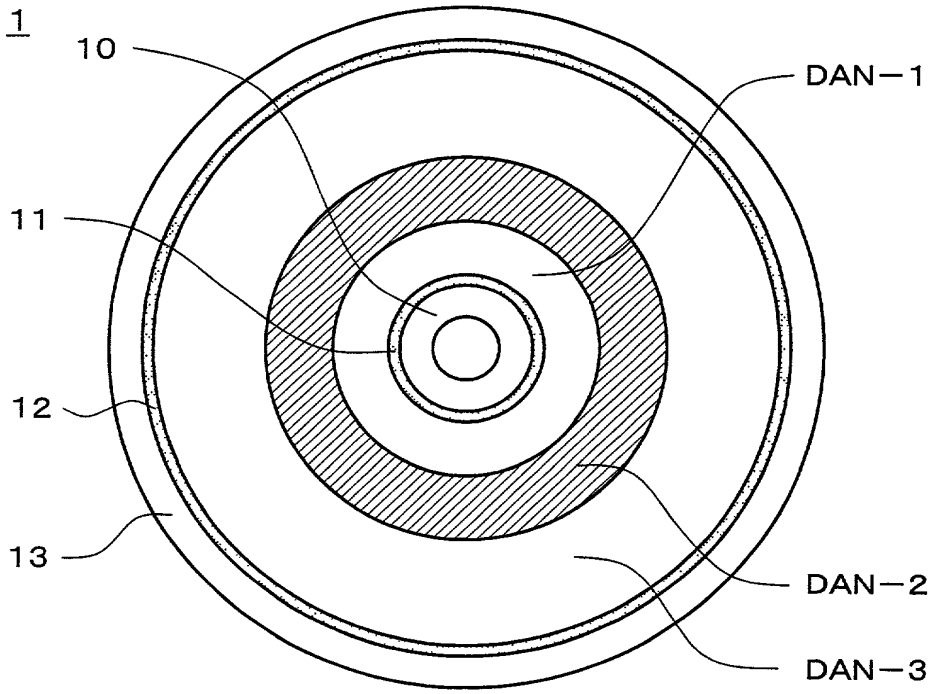
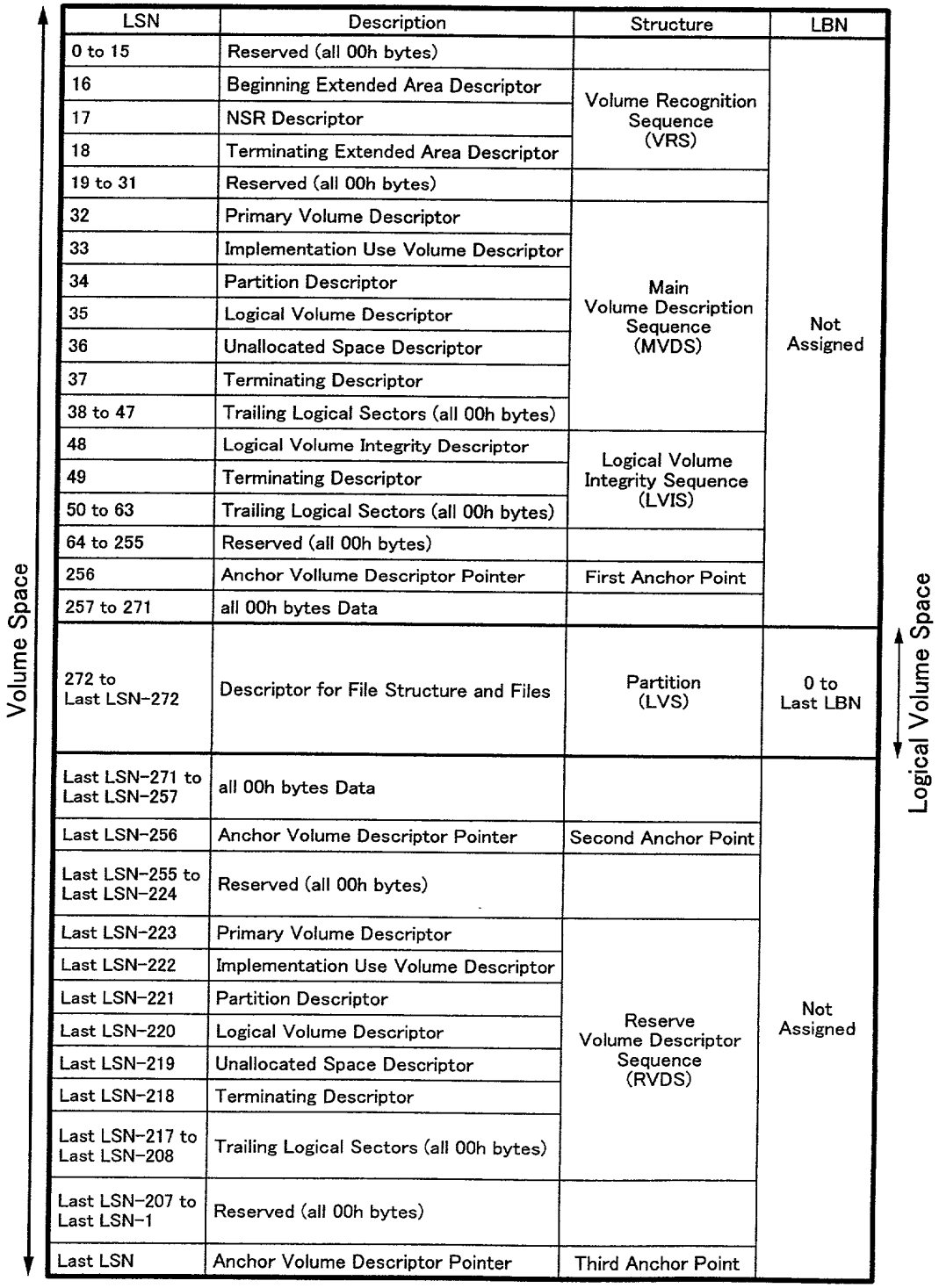


Fig. 15



LSN	Description	Structure	LBN
0 to 15	Reserved (all 00h bytes)		Not Assigned
16	Beginning Extended Area Descriptor	Volume Recognition Sequence (VRS)	
17	NSR Descriptor		
18	Terminating Extended Area Descriptor		
19 to 31	Reserved (all 00h bytes)		
32	Primary Volume Descriptor	Main Volume Description Sequence (MVDS)	
33	Implementation Use Volume Descriptor		
34	Partition Descriptor		
35	Logical Volume Descriptor		
36	Unallocated Space Descriptor		
37	Terminating Descriptor		
38 to 47	Trailing Logical Sectors (all 00h bytes)		
48	Logical Volume Integrity Descriptor	Logical Volume Integrity Sequence (LVIS)	
49	Terminating Descriptor		
50 to 63	Trailing Logical Sectors (all 00h bytes)		
64 to 255	Reserved (all 00h bytes)		
256	Anchor Vollume Descriptor Pointer	First Anchor Point	
257 to 271	all 00h bytes Data		
272 to Last LSN-272	Descriptor for File Structure and Files	Partition (LVS)	0 to Last LBN
Last LSN-271 to Last LSN-257	all 00h bytes Data		Not Assigned
Last LSN-256	Anchor Volume Descriptor Pointer	Second Anchor Point	
Last LSN-255 to Last LSN-224	Reserved (all 00h bytes)		
Last LSN-223	Primary Volume Descriptor	Reserve Volume Descriptor Sequence (RVDS)	
Last LSN-222	Implementation Use Volume Descriptor		
Last LSN-221	Partition Descriptor		
Last LSN-220	Logical Volume Descriptor		
Last LSN-219	Unallocated Space Descriptor		
Last LSN-218	Terminating Descriptor		
Last LSN-217 to Last LSN-208	Trailing Logical Sectors (all 00h bytes)		
Last LSN-207 to Last LSN-1	Reserved (all 00h bytes)		
Last LSN	Anchor Volume Descriptor Pointer	Third Anchor Point	

Fig. 16

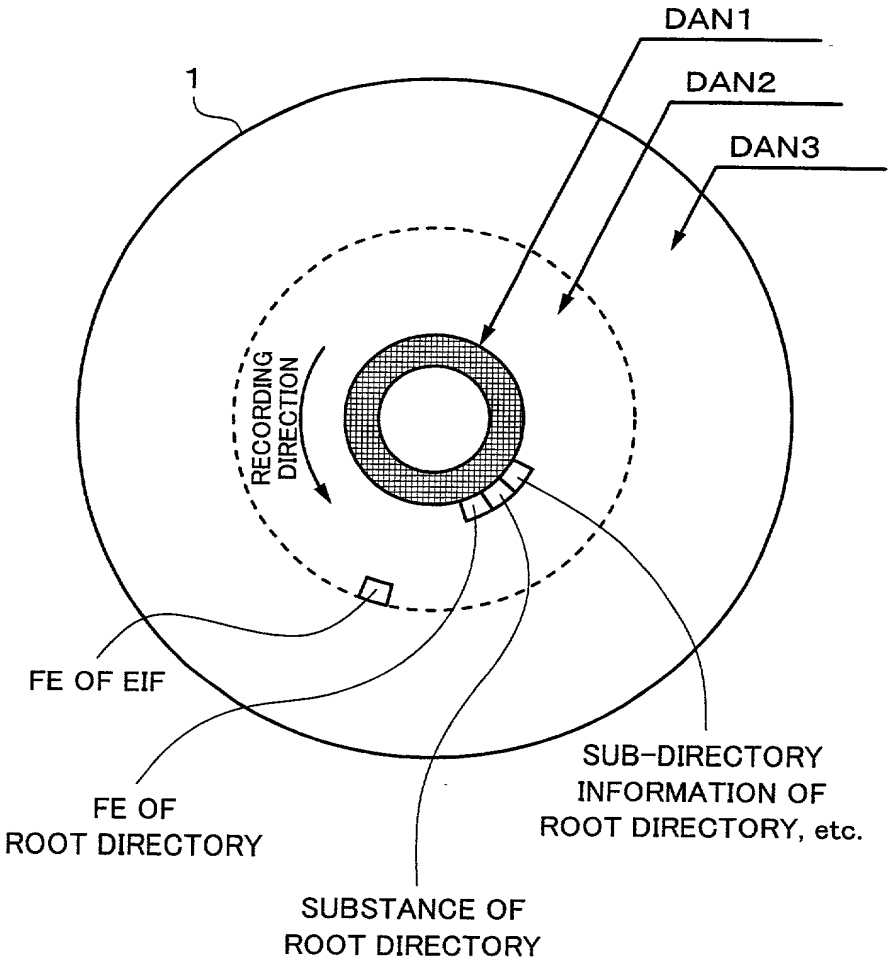

DAN1
DAN2
DAN3
1
RECORDING DIRECTION
FE OF EIF
FE OF ROOT DIRECTORY
SUB-DIRECTORY INFORMATION OF ROOT DIRECTORY, etc.
SUBSTANCE OF ROOT DIRECTORY

Fig. 17

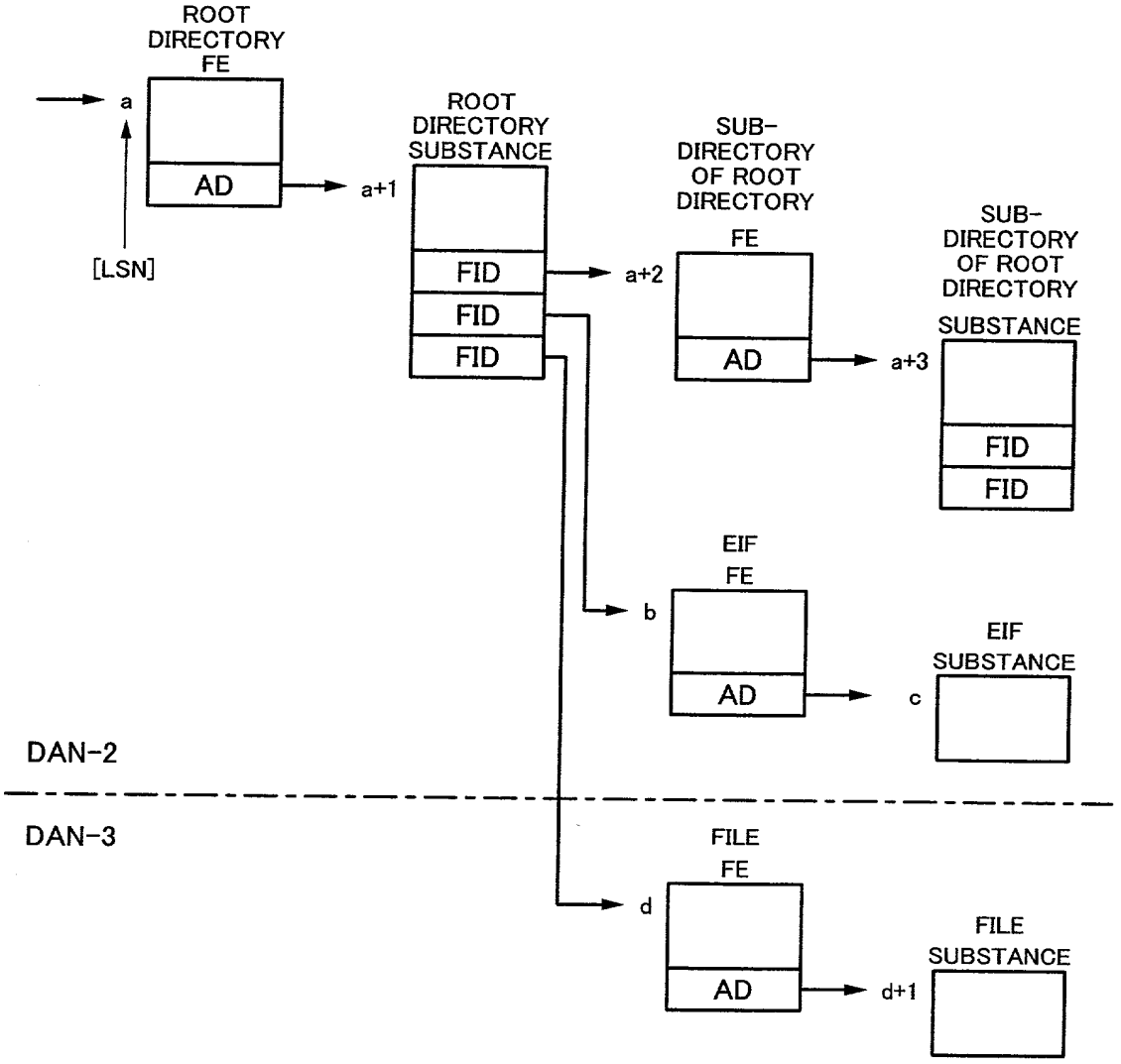

ROOT DIRECTORY FE
a
AD
[LSN]
a+1
ROOT DIRECTORY SUBSTANCE
FID
FID
FID
a+2
SUB-DIRECTORY OF ROOT DIRECTORY FE
AD
a+3
SUB-DIRECTORY OF ROOT DIRECTORY SUBSTANCE
FID
FID
b
EIF FE
AD
c
EIF SUBSTANCE
DAN-2
DAN-3
d
FILE FE
AD
d+1
FILE SUBSTANCE

Fig. 18